Journal of Combinatorial Theory, Series A **91**, 451–463 (2000)
doi:10.1006/jcta.2000.3110, available online at http://www.idealibrary.com on IDEAL®

The Umbral Transfer-Matrix Method. I. Foundations

Doron Zeilberger[1]

Department of Mathematics, Temple University, Philadelphia, Pennsylvania 19122
E-mail: zeilberg@math.temple.edu, http://www.math.temple.edu/~zeilberg/

Communicated by the Managing Editors

Received March 30, 2000

DEDICATED TO THE MEMORY OF GIAN-CARLO ROTA

subsequent parts of this series of articles, emerges when it is implemented on a computer. © 2000 Academic Press

Priests and Prophets

According to *cohen venavi*, a classic Hebrew essay by Ekhad Ha-Am, there are two ways of serving God: *Priest* and *Prophet*. The priest is a very skilled technician who knows by heart the 613 *mitzvahs*, and who can slaughter a lamb with due regard to the many rigorous and subtle laws. On the other hand, a prophet is often very clumsy in the day-to-day ritual, and is unable to sacrifice an animal properly. But prophets have a direct line to God. In the long run, their impact and influence far surpass those of the priests. Do we not all know the names of Isaiah, Jeremiah, and Ejekiel? Yet none of us remembers any of the names of the high priests who lived at the same time.

Gian-Carlo Rota was a paragon of the mathematical prophet. His technical contributions, while substantial, dwarf in comparison to his *vision*, *insight*, and new unifying *concepts*. Also, his uncanny *realizations* that some things are *important* have revolutionized more than one combinatorial area.

Rota's dislike of routine priestly work is expressed nicely in the following extract from Richard Stanley's touching and warm obituary that appeared in *SIAM News*:

[1] Supported in part by the NSF.

0097-3165/00 $35.00

> ROTA was always interested in the "BIG picture" and trying to understand the true essence of any subject in which he was interested.

Conversely, Rota was not interested in the "little picture" and, as far as I know, never solved any major (or minor) specific open problem. He was often criticized for publishing "trivial" results, for example, "using 50 pages to prove the Vandermonde–Chu identity." It is very possible that some of his longer papers, especially those about the Umbral Calculus, would not have found a journal, had they not been accepted to *Advances in Mathematics.*

Luckily, Gian-Carlo was unfazed by these remarks. He was a GREAT GURU, and fortunately, he knew it. In one of his numerous inimitable *Forewords* (this one to "Species," by F. Bergeron, G. Labelle, and P. Leroux), he wrote:

> There is a second way in which mathematics advances. It happens whenever some common sense notion that had heretofore been taken for granted is discovered to be wanting, to need clarification and definition.

While he was talking here about André Joyal's *species*, I am sure that he was also referring to his own numerous *clarifications of common sense notions.* In particular that of the *umbra.*

The *Umbral Calculus* was a standard tool of the trade in 19th-century algebra, but it was always surrounded by a magical aura, and the only attempt at a full rigorization before Rota, by E. T. Bell, was a conceptual flop. Rota's stroke of genius had to wait for the 20th century and for the notion of the *linear functional.* Indeed Rota's seminal contribution, which, like almost all major breakthroughs, is *a posteriori* obvious, was the *mere REALIZATION* that an umbra *is* a linear functional.

Let us recall Rota's favorite example. Prove that

$$a_n = \sum_{k=0}^{n} \binom{n}{k} b_k, \tag{1}$$

if and only if

$$b_n = \sum_{k=0}^{n} (-1)^{n-k} \binom{n}{k} a_k. \tag{1'}$$

The classical umbral proof goes as follows. Let $a = b + 1$. Then by the binomial theorem

$$a^n = \sum_{k=0}^{n} \binom{n}{k} b^k, \tag{2}$$

and also, since $b = a - 1$,

$$b^n = \sum_{k=0}^{n} (-1)^{n-k} \binom{n}{k} a^k. \tag{2'}$$

Now just "lower the superscripts (powers) and make them subscripts."

Rota's rendition is as follows. Define a *linear functional* ("umbra") A on the vector space of polynomials, by defining it on monomials by $A(x^n) := a_n$, and extending by linearity. Similarly define $B(x^n) := b_n$. Equation (1) says that $A(x^n) = B((x+1)^n)$ for $n = 0, 1, 2, \ldots$. By linearity, $A(p(x)) = B(p(x+1))$, for *any* polynomial $p(x)$, and hence $B(q(x)) = A(q(x-1))$ for *any* polynomial $q(x)$. In particular, if $q(x) = x^n$, we get (1′).

Rota's insight led to a beautiful theory [R, RR, LR], and enabled a redoing of classical invariant theory in the "right" way [KR]. It also led to many new results in the theory that did not make sense before. But, as many narrow-minded petit priests argued, "it was not good for anything."

In this series of papers, I hope to show how Rota's beautiful concept, coupled with the Transfer-Matrix Method, could be used to COMPUTE generating functions for many hard-to-count combinatorial objects, like certain important subsets and supersets of lattice animals and self-avoiding walks.

Hamming Got It Backwards: The Purpose of Insight Is Computation

Speaking of *computations*, many mathematicians feel the need to apologize for computing, and Hamming's famous quip, "*the purpose of computation is not numbers but insight*," has deteriorated into a cliché. This is a hold-over of our *pure*-itan upbringing, which made us feel guilty about "computation without insight" the same way our parents used to feel guilty about "sex without love."

There is nothing wrong with *computation for computation's sake*. I would love to know the number of self-avoiding walks with 200 steps, or the number of polyominoes with 200 cells. Very often the numerical output itself does not give any new "insight," yet the attempt to compute it does, but even if it does not, there is still nothing wrong with brute computation.

Actually, there is! Completely brutish computations cannot be carried very far. So at present we need INSIGHT to do *efficient* computations. In the future, that insight may very well come from computers themselves, but at present, we still need humans like Rota to supply such insight.

Most mathematicians will soon be replaceable by machines (some of us already are); yet prophets like Gian-Carlo will, most probably, *always* be needed.

Why Is the Concept of Umbra So Crucial?

I will soon define the *Umbral Transfer-Matrix Method*, but beforehand let me explain, in general terms, why the notion of the umbra is *so* important. In the finite transfer-matrix method [Z0], the entries of the matrix are numbers or polynomials. In the current theory, the entries are *operators*. In all but the most trivial applications, the operators are very complicated, and it would be impossible to find them by hand. So we need the computer to "do research" and find the operators. But an operator is a rather abstract notion, so how can a "dumb" computer find it? It cannot reason combinatorially. It turns out that, in many cases of interest, it is possible to *mechanize* the action of the operator on a generic monomial, since as will become clear from the examples below, it only involves summing (finite and infinite) geometrical series, or their derivatives, which Maple does very well, and adding them up, which Maple does equally well. Then, since we know what kind of operator to expect, it is possible to *automatically deduce* the operator from its action on a generic monomial, and hence Maple can write down the operator (in a formal, Maple-readable, format). Once we have the Operator-Matrix, we can use it to generate, in *polynomial time*, series expansions for the combinatorial objects considered, and sometimes even to solve the induced system of mixed differential-q-equations *explicitly*, enabling us to obtain the generating function in some kind of *closed form*.

Weighted Finite-Parameter Infinite Directed Graphs

The Finite Transfer-Matrix Method is used to weight-enumerate paths on a *finite* digraph; see [Z0] for a detailed exposition. Here, we will be considering directed graphs whose set of vertices, V, is *infinite*, with possibly multiple edges. Even though there are infinitely many vertices, we will assume that they can be partitioned into a finite union of *vertex-families*, $\{v_1, ..., v_n\}$, such that each family v_i is an l_i-parameter infinite family, parameterized by the l_i discrete variables $(a_1, ..., a_{l_i})$, where $(a_1, ..., a_{l_i})$ ranges over a well-defined subset D_i of $\{0, 1, 2, 3, ...\}^{l_i}$. So the vertex set of our infinite digraph can be partitioned as follows:

$$V = \bigcup_{i=1}^{n} \bigcup_{(a_1, ..., a_{l_i}) \in D_i} v_i(a_1, ..., a_{l_i}).$$

We will also assume that for any pair of vertex types v_i and v_j, there are $K(i, j) \geqslant 0$ families of edges, and for each of $k = 1, ..., K(i, j)$, the type-k edge coming out of vertex $v_i(a_1, ..., a_{l_i})$ may wind up in any of the vertices $v_j(b_1, ..., b_{l_j})$, where $(b_1, ..., b_{l_j})$ may belong to a well-defined subset of D_j;

let us call it $E_{i,j}^{(k)}(a_1, ..., a_{l_i})$. We also assume that every such edge has a certain *weight* given by a *weight-function*

$$W_{i,j}^{(k)}(a_1, ..., a_{l_i}; b_1, ..., b_{l_j}).$$

The weight of a *path P*, $Wt(P)$, is the sum of the weights of its participating edges. We are interested in computing the weight-enumerator of all paths

$$\sum_{P \in \text{Paths}} q^{Wt(P)},$$

either explicitly, or, failing this, by using a polynomial-time algorithm for computing the *series expansion*, i.e., the first N terms of its power-series expansion, for any given N.

We now digress to define a *Rota Operator.*

Definition of an Atomic Rota Operator. An atomic Rota operator from the ring of formal power-series in r variables $Z(q)(x_1, ..., x_r)$ to the ring of formal power-series in s variables $Z(q)(y_1, ..., y_s)$ (with coefficients from the ring of integer-coefficient formal power-series in q) is an operator of the form

$$T[f(x_1, ..., x_r)] = R(q, y_1, ..., y_s)\, D_{x_1}^{\alpha_1} \cdots D_{x_r}^{\alpha_r} f(x_1, ..., x_r)\,|_{\{x_1 = m_1, ..., x_r = m_r\}}, \tag{ARO}$$

where $R(x, q_1, ..., x_r)$ is a rational function of all its arguments; $D_{x_1}, ..., D_{x_r}$ are the differentiation operators with respect to $x_1, ..., x_r$, respectively; $\alpha_1, ..., \alpha_r$ are non-negative integers; and $m_1, ..., m_r$ are each *monomials* in the variables $(q, y_1, ..., y_s)$.

An Example of an Atomic Rota Operator.

$$f(x_1, x_2) \rightarrow \frac{q^3 y_1 y_2 y_3}{(1 - q y_1)(1 - q^2 y_1 y_2 y_3)} D_{x_1} D_{x_2}^3 f(y_1 y_2 y_3, q y_3).$$

Maple Representation of Atomic Rota Operators. The Atomic Rota Operator of (ARO) is represented in our Maple packages by a list of length 3:

$$[R, [\alpha_1, ..., \alpha_r], [m_1, ..., m_r]].$$

For example: the above operator, in Maple, would read

```
[q ** 3 * y1 * y2 * y3/((1 - q * y1)
      * (1 - q ** 2 * y1 * y2 * y3)), [1, 3], [y1 * y2 * y3, q * y3]].
```

Definition of a Rota Operator. A *Rota operator* is a sum of atomic Rota operators.

Maple Representation of Rota Operators. The Rota operator $\mathscr{P} = \mathscr{P}_1 + \cdots + \mathscr{P}_m$, where $\mathscr{P}_1, ..., \mathscr{P}_m$ are atomic Rota operators, is denoted by the set $\{P_1, ..., \mathscr{P}_m\}$. For example, the Rota operator $f(x) \to qf'(1) + xf(qx) + (x/(1-qx)\, f''(q^3)$ is represented by $\{[q, [1], [1]], [x, [0], [q * x]], [x/(1-q*x), [2], [q^3]]\}$.

It turns out that in many applications, the following property holds:

The Umbral Axiom

For every pair of *vertex types*, v_i, v_j, and for each of its $K(i, j)$ edge types connecting them, the following operator from $Z(q)(x_1, ..., x_{l_i})$ to $Z(q)(y_1, ..., y_{l_j})$, defined on the basis of monomials by

$$x_1^{a_1} \cdots x_{l_i}^{a_{l_i}} \to \sum_{(b_1, ..., b_{l_j}) \in E^k_{i,j}(a_1, ..., a_{l_i})} q^{W^{(k)}_{i,j}(a_1, ..., a_{l_i}; b_1, ..., b_{l_j})} y_1^{b_1} \cdots y_{l_j}^{b_{l_j}},$$

is an atomic Rota operator; let us call it $Q^k_{i,j}$.

Also, let us define the *transition-operator* from vertices of type i to vertices of type j $(1 \leqslant i, j \leqslant n)$ by

$$\mathscr{P}_{i,j} := \sum_{k \in K(i,j)} Q^k_{i,j},$$

which by our assumption are all Rota operators.

Let us define the *mishkal* of a path P, in our digraph, that ends with the vertex $v_i(a_1, ..., a_{l_i})$, by

$$q^{Wt(P)} x_1^{a_1} \cdots x_{l_i}^{a_{l_i}},$$

and let us define the *total mishkal* of all the paths that end in a type-i vertex by

$$F_i(q; x_1, ..., x_{l_i}) := \sum_P \mathit{mishkal}(P),$$

where the sum extends over the infinite set of paths that end in a type-i vertex.

It follows immediately from this set-up that the n formal power-series F_j $(j = 1, ..., n)$ satisfy the following system of n differential–functional equations,

$$F_j = [\, j \in \mathit{Start}\,] + \sum_{i=1}^{n} \mathscr{P}_{i,j} F_i. \qquad (\mathit{FundamentalSystem})$$

In the lucky case, we can solve this system explicitly, but at any rate, we can use it iteratively to find a series expansion in q. In either case, the desired weight-enumerator is given by

$$\sum_{j \in Finish} F_j(q; 1, ..., 1).$$

Note that the variables $x_1, ..., x_{l_i}$ corresponding to the l_i-parameter vertex type i, for $i=1, ..., n$, serve as *catalysts*, all to be discarded (i.e., substituted by 1) at the end of the "reaction."

At this point, all this sounds like very abstract nonsense. We hope that the following simple examples will make the new concept clearer. In subsequent parts [Z2–Z5], we hope to present "heavy-duty" examples that would be hopeless without a computer.

Vertex-Weighted Infinite Directed Graphs

Even though it is easy to subsume this case in the former, edge-weighted, case, it is pedagogically, and implementation-wise, easier to treat it separately. In this model, the vertices rather than the edges are endowed with weights, $wt(v_i(a_i))$, and the weight of a path is the sum of the weights of its vertices (counted separately for each visit). Here we also assume that there are $K_{i,j}(a_i, b_j)$ edges between vertices $v_i(a_i)$ and $v_j(b_j)$.

The Umbral Axiom now takes the form

The Umbral Axiom for Vertex-Weighted Digraphs

For every pair of *vertex types*, v_i, v_j, the operator from $Z(q)(x_1, ..., x_{l_i})$ to $Z(q)(y_1, ..., y_{l_j})$, defined on the basis of monomials by

$$x_1^{a_1} \cdots x_{l_i}^{a_{l_i}} \to \sum_{j=1}^{n} \sum_{(b_1, ..., b_{l_j}) \in D_j} K_{i,j}(a_i, b_j)\, q^{W_j(b_1, ..., b_{l_j})} y_1^{b_1} \cdots y_{l_j}^{b_{l_j}},$$

is a Rota operator; let us call it $\mathscr{P}_{i,j}$.

We also assume that, for all vertex types $i=1, ..., n$ the formal power-series

$$I_i(q, x_1, ..., x_{l_i}) := \sum_{(a_1, ..., a_{l_i}) \in D_i} q^{W_i(a_1, ..., a_{l_i})} x_1^{a_1} \cdots x_{l_i}^{a_{l_i}},$$

of 1-*vertex* paths, are all rational functions. The analog of (*FundamentalSystem*) for vertex-weighted digraphs is

$$F_j = [\, j \in Start\,]\, I_j + \sum_{i=1}^{n} \mathscr{P}_{i,j} F_i \qquad (FundamentalSystemVW)$$

Both (*FundamentalSystem*) and (*FundamentalSystemVW*), have the form of an

Umbral Scheme

$$F_i = A_i + \sum_{j=1}^{n} \mathcal{Q}_{i,j} F_j, \qquad (UmbralScheme)$$

where $F_j(q, x_1, ..., x_{l_j})$ are the unknown formal-power series, and $\mathcal{Q}_{i,j}$ are explicit Rota operators, together with a subset of $\{1, 2, ..., n\}$, S, and the desired quantity is the formal power-series

$$\sum_{i \in S} F_i(q, 1, ..., 1),$$

where in the argument of F_i, there are l_i 1's.

Several Statistics. Suppose that we want to weight-enumerate according to several attributes (statistics); then all we said above goes verbatim, except that q is replaced by a multi-variable $\mathbf{q}$.

Maple Representations of Umbral Schemes. We will represent the Umbral Scheme (*UmbralScheme*) as a list of length 4,

$$[S, UmbralMatrix, InitialVector, VariablesLists],$$

where *UmbralMatrix* is the matrix of Rota operators

$$UmbralMatrix = [[\mathcal{Q}_{1,1}, \mathcal{Q}_{1,2}, ..., \mathcal{Q}_{1,n}], ..., [\mathcal{Q}_{n,1}, \mathcal{Q}_{n,2}, ..., \mathcal{Q}_{n,n}]].$$

In particular, the number of elements of *UmbralMatrix*, n, determines the number of vertex types, which we will assume are called $\{1, 2, ..., n\}$. S is the subset of $\{1, 2, ..., n\}$ mentioned above.

InitialVector is the vector of rational functions

$$InitialVector = [A_1, ..., A_n].$$

Finally, *VariablesLists* is the list of variables that the F_i depend on (not counting q):

$$VariablesLists = [[x_1, ..., x_{l_1}], [x_1, ..., x_{l_2}], ..., [x_1, ..., x_{l_n}]].$$

Note that we may use $x_1, x_2, ...$ for each of the argument sets of F_i $(i = 1, ..., n)$; i.e., we do not have to invent new names, since these variables are local.

Very Simple Examples That Even Humans Can Work Out

The *real power* of the new concept would only emerge in cases that *only* computers can do. But since we still need humans to program the computer, the programmer (in this case myself, but I hope to be joined by others who will use this method in the future) would need to really understand and internalize the concept, before he or she (and soon, also, it) could write the software. The best way to understand new concepts and methods is to work out a few examples by hand. These examples could also serve as test cases for the computer programs.

EXAMPLE 1. *Ordinary Partitions.* Here the digraph consists of one vertex type, parameterized by positive integers $v(a)$, with weight a. A partition $a_1 \leqslant a_2 \leqslant \cdots \leqslant a_k$ can be represented as a path $v(a_1) \to \cdots \to v(a_k)$. Let us call the catalytic variable x; the *mishkal* of such a path is then $q^{a_1+\cdots a_k} x^{a_k}$. Out of $v(b)$ we may go to any $v(a)$ with $a \geqslant b$. Since we only have here one vertex type, the umbral matrix is a 1×1 matrix $(\mathscr{P}_{1,1})$, and the operator $\mathscr{P}_{1,1}$ acts on monomials x^b by

$$x^b \to \sum_{a \geqslant b} (qx)^a = \frac{(qx)^b}{1-qx},$$

which, extended linearly, implies that

$$\mathscr{P}_{1,1} f(x) = \frac{f(qx)}{1-qx}.$$

The Umbral Scheme is thus

$$f(x) = \frac{qx}{1-qx} + \frac{f(qx)}{1-qx},$$

which, in this simple case, immediately implies that $g(x) = 1 + f(x)$ satisfies

$$g(x) = \frac{g(qx)}{1-qx},$$

and hence

$$g(x) = \prod_{i=1}^{\infty} \frac{1}{1-q^i x},$$

and finally, setting the catalyst variable, x, to 1, we get a new proof of Euler's generating function for integer-partitions, $g(1) =$

$$\prod_{i=1}^{\infty} \frac{1}{1-q^i}.$$

EXAMPLE 2. *2-Rowed Plane Partitions.* Recall that a 2-rowed plane-partition, with r columns, is a $2\times r$ array of non-negative integers

$$\begin{matrix} a_{1,1} & a_{1,2} & \cdots & a_{1,r}, \\ a_{2,1} & a_{2,2} & \cdots & a_{2,r}, \end{matrix}$$

such that $a_{i,1}\geqslant a_{i,2}\geqslant \cdots \geqslant a_{i,r}\geqslant 0$, for $i=1, 2$, and $a_{1,j}\geqslant a_{2,j}\geqslant 0$, for $j=1, ..., r$. Note that we allow 0, so that even the all-zero matrix, for example, is being counted. The weight of a plane partition is the sum of its entries, and in order to keep track of the number of columns, we will use the letter t. By (a special case of) MacMahon's box theorem, the weight-enumerator of 2-rowed plane-partitions,

$$\sum_{A} q^{|A|} t^{\#\mathit{columns}(A)},$$

where the sum ranges over all 2-rowed plane partitions, equals

$$\sum_{r=0}^{\infty} \frac{(1-q)}{(q)_r\,(q)_{r+1}}\, t^r. \qquad (Percy)$$

Using the present approach, we should be able to rederive the full expansion of (*Percy*), but at this stage we only want to illustrate the concept, so we will be content to show how Maple can generate the first R terms of (*Percy*), for any specified R.

Now the digraph still only consists of one vertex type, but it is parameterized by *ordered pairs* of integers, (a_1, a_2), where $a_1\geqslant a_2\geqslant 0$. From a vertex $v(b_1, b_2)$ we may go to $v(a_1, a_2)$, satisfying $a_1\geqslant a_2\geqslant 0$, and $a_1\geqslant b_1, a_2\geqslant b_2$. So there is a natural one-to-one correspondence between r-vertex paths on this infinite digraph, and $2\times r$ plane partitions. Let us call the catalytic variables x_1, x_2. Again, we only have here one vertex type, so the umbral matrix is a 1×1 matrix $(\mathscr{P}_{1,1})$, and the operator $\mathscr{P}_{1,1}$ acts on monomials $x_1^{b_1}x_2^{b_2}$ by

$$\begin{aligned} x_1^{b_1}x_2^{b_2} \to t & \sum_{\substack{a_1\geqslant a_2\geqslant 0,\\ a_1\geqslant b_1,\, a_2\geqslant b_2}} (qx_1)^{a_1}\,(qx_2)^{a_2} \\ &= t\sum_{a_2=b_2}^{b_1-1}\sum_{a_1=b_1}^{\infty} (qx_1)^{a_1}\,(qx_2)^{a_2} + t\sum_{a_2=b_1}^{\infty}\sum_{a_1=a_2}^{\infty} (qx_1)^{a_1}\,(qx_2)^{a_2} \\ &= t\,\frac{(qx_1)^{b_1}}{1-qx_1}\left[\frac{(qx_2)^{b_2}-(qx_2)^{b_1}}{1-qx_2}\right] + \frac{t}{1-qx_1}\sum_{a_2=b_1}^{\infty} (qx_2)^{a_2}\,(qx_1)^{a_2} \\ &= t\,\frac{(qx_1)^{b_1}\,(qx_2)^{b_2}-(qx_1)^{b_1}\,(qx_2)^{b_1}}{(1-qx_1)(1-qx_2)} + t\,\frac{(q^2x_1x_2)^{b_1}}{(1-qx_1)(1-q^2x_1x_2)}. \end{aligned}$$

Extending by linearity, we see that $\mathscr{P}_{1,1}$ is the umbra

$$\mathscr{P}_{1,1} f(x_1, x_2) = t\,\frac{f(qx_1, qx_2) - f(q^2x_1x_2, 1)}{(1-qx_1)(1-qx_2)} + t\,\frac{f(q^2x_1x_2, 1)}{(1-qx_1)(1-q^2x_1x_2)}.$$

The Umbral Scheme is thus

$$F(x_1, x_2) = \frac{t}{(1-qx_1)(1-q^2x_1x_2)} + t\,\frac{F(qx_1, qx_2) - F(q^2x_1x_2, 1)}{(1-qx_1)(1-qx_2)} + t\,\frac{F(q^2x_1x_2, 1)}{(1-qx_1)(1-q^2x_1x_2)}. \qquad (UMP2)$$

Starting with $F = t/(1-qx_1)(1-q^2x_1x_2)$, and iteratively plugging into the right side of $(UMP2)$, and expanding with respect to increasingly higher powers of t, would yield the MacMahon expansion $(Percy)$ to any desired "accuracy."

EXAMPLE 3. *Counting Compositions without Double Descents.* We would like to find the generating function

$$\sum_{n=0}^{\infty} A(n)\, q^n,$$

where $A(n)$ is the number of vectors of positive integers $(a_1, ..., a_r)$ (where $r \geqslant 0$), such that $a_1 + \cdots + a_r = n$ and we are not allowed to have a double descent $a_i > a_{i+1} > a_{i+2}$ for any $i = 1, ..., r-2$. For example (4, 5, 6, 4, 4, 6, 5, 7) is allowed, but (4, 5, 6, 4, 2, 5, 6, 4) is not allowed.

To model this as paths on a digraph, we have to introduce two kinds of vertices u (up) corresponding to the situation where the entry before it is smaller, including the case when it is at the very beginning; and vertices d (down) corresponding to the situation where the entry before it is bigger. In the digraph, the followers of $u(b)$ are $u(a)$ with $a \geqslant b$, as well as $d(a)$ with $1 \leqslant a < b$. The followers of $d(b)$ may only be $u(a)$ with $a \geqslant b$.

Hence the operator $\mathscr{P}_{u,u}$ acts on a monomial by

$$x^a \to \sum_{b=a}^{\infty} (qx)^b = \frac{(qx)^a}{1-qx},$$

the operator $\mathscr{P}_{u,d}$ acts on a monomial by

$$x^a \to \sum_{b=1}^{a-1} (qx)^b = \frac{qx - (qx)^a}{1-qx},$$

the operator $\mathscr{P}_{d,u}$ acts on a monomial by

$$x^a \to \sum_{b=a}^{\infty} (qx)^b = \frac{(qx)^a}{1-qx},$$

while $\mathscr{P}_{d,d}$ is 0. By linearity, the operators extend to any formal power series in x by

$$P_{u,u}[f(x)] = \frac{f(qx)}{1-qx},$$

$$P_{u,d}[f(x)] = \frac{qxf(1) - f(qx)}{1-qx},$$

$$P_{d,u}[f(x)] = \frac{f(qx)}{1-qx},$$

$$P_{d,d}[f(x)] = 0.$$

Let $F_u(x)$ and $F_d(x)$ be as above, where they also depend on q, of course. The Umbral Scheme is

$$F_u(x) = \frac{qx}{1-qx} + \frac{F_u(qx)}{1-qx} + \frac{F_d(qx)}{1-qx},$$

$$F_d(x) = \frac{qxF_u(1) - F_u(qx)}{1-qx}.$$

The desired quantity is $F_d(1) + F_u(1)$.

ROTA: The Accompanying Maple Package

The Maple package `ROTA` is available from the web page of this series of articles,

`http://www.math.temple.edu/~zeilberg/utm.html.`

The main procedure is `ApplyUmSc(UmSch,q,n,vars)`. It inputs an umbral scheme, `UmSch`, a variable `q`, and an integer `n`, and a set of variables, `vars` (the catalytic variables). It outputs the series expansion, in the variable `q`, up to the term `q**n` of the computed generating function, both with the catalytic variables retained and with them made 1. For example if the umbral scheme is

$$UMP1 := [\{1\}, \{1\}, [[\{[1/(1-q*x), [0], [q*x]]\}]], [q*x/(1-q*x)], [[x]]]:$$

then `ApplyUmSc(UMP1,q,7,{x})[2];` yields `[1,2,3,5,7,11,15]`.

ROTA also contains five sample umbral schemes: UMP1 (displayed just above); UMP2, featured in Example 2 above; UMW, which counts compositions with no double descents (Example 3 above); as well as UmSch1 and UmSch2, to be discussed in [Z3].

Future Plans. I hope to apply the Umbral Transfer-Matrix Method in forthcoming sequels to the present article [Z2–Z5].

Final Remark. The scenario in [Z] can be viewed as a very special case of the present set-up, namely, the case where the monomials $m_1, ..., m_r$ featured in the definition of an atomic Rota operator are all equal to 1. As was shown in [Z], in this case the generating function is always computable explicitly, and furthermore is a rational function.

REFERENCES

[KR] J. P. S. Kung and G.-C. Rota, The invariant theory of binary quadratic forms, *Bull. Amer. Math. Soc. (N.S.)* **10** (1984), 27–85.

[LR] D. E. Loeb and G.-C. Rota, Formal power series of logarithmic type, *Adv. in Math.* **75** (1989), 1–118.

[RR] S. M. Roman and G.-C. Rota, The umbral calculus, *Adv. in Math.* **27** (1978), 95–188.

[R] G.-C. Rota, The number of partitions of a set, *Amer. Math. Monthly* **71** (1964), 498–504. Reprinted in "Gian-Carlo Rota on Combinatorics" (J. P. S. Kung, Ed.), pp. 111–117, Birkhaüser, Basel, 1995.

[Z] D. Zeilberger, Automated counting of Lego Towers, *J. Difference Equations Appl.* **5** (1999), 323–334.

[Z0] D. Zeilberger, SYMBOL-CRUNCHING with the TRANSFER-MATRIX Method in order to count SKINNY physical creatures, preprint. [Available from Zeilberger's website]

[Z1] D. Zeilberger, The Umbral Transfer-Matrix Method. I. Foundations, *J. Combin. Theory Ser. A* **91** (2000), doi:10.1006/jcta.2000.3110.

[Z2] D. Zeilberger, The Umbral Transfer-Matrix Method. II. Counting plane and solid partitions, in preparation.

[Z3] D. Zeilberger, The Umbral Transfer-Matrix Method. III. Counting animals, in preparation.

[Z4] D. Zeilberger, The Umbral Transfer-Matrix Method. IV. Counting self-avoiding polygons and walks, in preparation.

[Z5] D. Zeilberger, The Umbral Transfer-Matrix Method. V. The Goulden–Jackson Cluster Method for infinitely many mistakes, in preparation.